uRADMonitor SMOGGIE

Low Cost Environmental Monitoring

Temperature, Barometric pressure, Relative Humidity, Particulate matter (PM1.0, PM2.5, PM10)

Features

- 2 high quality sensors tracking 6 air parameters
- Particulate Matter laser scattering counter
- Open source hardware & software
- Arduino compatible
- Integrated WiFi Internet connectivity
- USB port for power, debug and configuration
- Built-in air pump for active flow
- Direct and Cloud data access via API
- Low power consumption
- Ultra low cost

Applications

- Home monitoring
- Citizen science
- Smart cities
- IOT / Internet of things

Description

The traditional balance between performance and cost is a thing of the past with the new uRADMonitor SMOGGIE device, designed for mass deployment to generate block level particulate matter readings. The "Smoggie" is a smog fighter with it high performance Particulate Matter sensor that uses laser scattering to count the individual particles.
With the built in Wifi Connectivity, the device will report all measurements to the uRADMonitor servers.
It is an ultra low cost device, where all its building blocks have been cost optimized except for one: its quality.

The design is open source, with complete hardware and software details publicly available on Github. It comes pre-programmed, but further modifications on its software are possible using Arduino. By default, all measurements are sent to the uRADMonitor servers, and are accessible with the API or can be viewed online. This makes it convenient for the classroom, for workshops or citizen science projects.

The uRADMonitor network is a global array of interconnected monitoring stations, focused on continuous Environmental Surveillance. Its purpose is to generate fully transparent open data, used to assert the quality of our environment. The uRADMonitor SMOGGIE data is accessible in real time via an API interface directly from the uRADMonitor cloud.

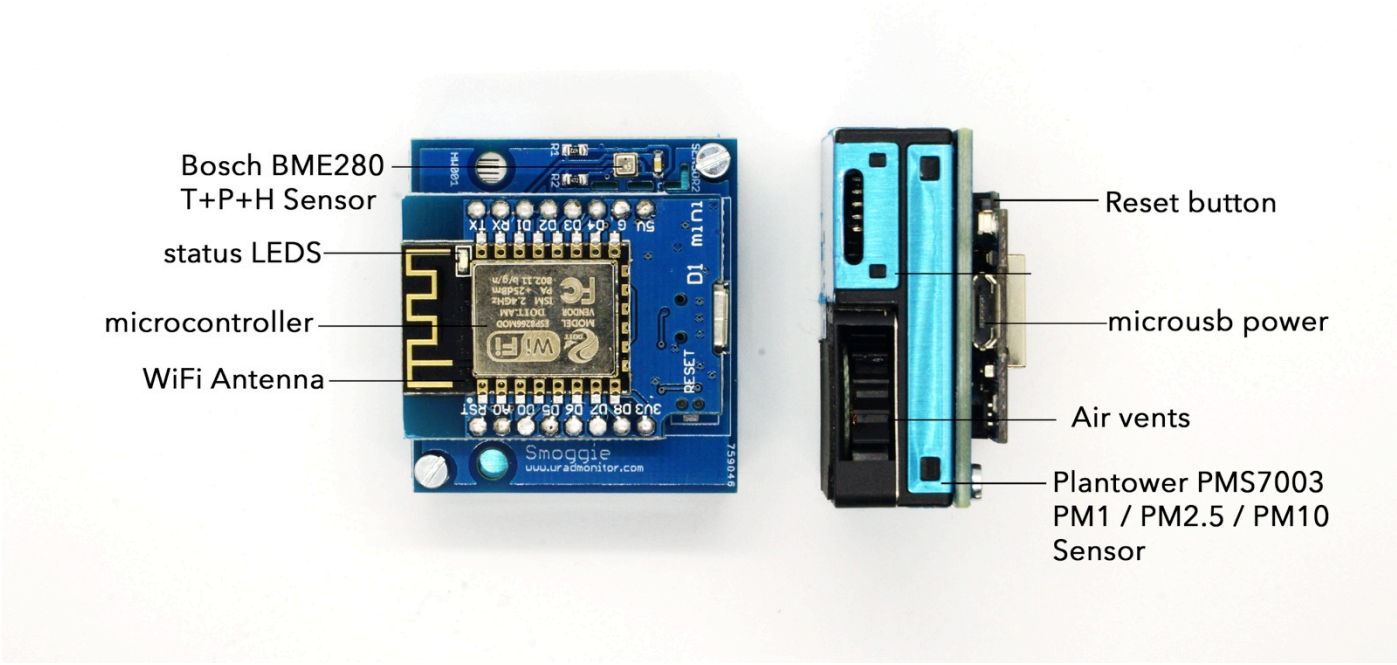
Sensors

The uRADMonitor SMOGGIE is designed as an ultra low cost IOT Environmental detector with a high precision Particulate Matter PM2.5 laser scattering sensor and the BME280 from Bosch for temperature, pressure, humidity. A built in fan assures an active air flow stream across the sensing elements. The device connects to your wireless Internet Router via WiFi, to send the readings online.

Sensor	Parameter	Minimum value	Maximum value	Absolute Accuracy
Bosch BM280	Temperature	-40 °C	+85 °C	± 1°C
	Pressure	300 hPa	1100 hPa	± 0.25 %
	Humidity	0% RH	100% RH	± 3 %
Plantower PMS7003	PM1.0 PM2.5 PM10	0 µg/m³	1000 µg/m³	± 15 %

Specification

Parameter	uRADMonitor AIR
Internet connection	WLAN connectivity to WiFi Internet Router
Standards	WLAN 2.4GHz IEEE 802.11 b/g/n
Wireless frequencies	2.400–2.4835 GHz ISM band
Modem Chip	Espressif esp8266
Modem certifications	CE, FCC
Antenna connector	PCB antenna
Enclosure Protection	IP30
Supply Voltage	micro USB 5V
Dimensions	35x35x20 mm (excl. sup and enclosure)
Weight	50g
Mounting	fixed
Recommended Use Ratings	Temperature: -20°C to +65°C Humidity: 0RH to 95RH



uRADMonitor SMOGGIE assembled circuit board diagram in hardware version 100

Usage guide

- **Power supply**

The SMOGGIE uses a standard micro USB connector that is used to power the unit with a regular phone charger. The unit takes 5V to run.

- **Outdoor use and exposure to elements**

The unit comes in a plastic enclosure that protects the sensitive electronics from the elements. It can be directly installed outdoors. Make sure the USB connector faces down, so no rain can get inside. Do not cover the air circulation holes.

- **Precautions**

Do not expose the device to a large amount of dust such as in the woodworking centers. Do not expose the appliance to solvents or to a large amount of concentrated vapors of chemicals (acetone, paints, alcohol, butane, propane, etc.), because the sensors can wear out, or the measurements may become inconclusive. Do not expose the apparatus to mechanical shocks. Wherever possible, mount the appliance in a vertical position to extend the life of the built-in fan mechanisms.

- **Installing the unit**

For mounting, use the hole in the housing bracket. Ensure that you properly connect the power cord and secure it against vibration where necessary.

Warranty

uRADMonitor SMOGGIE is covered by a 12 months warranty for any defects in material or workmanship, under normal use.

Data access

uRADMonitor is designed for easy and open data access. The data can be accessed in two ways:

- **Local access**

Applies where the uRADMonitor unit is part of a LAN network. The uRADMonitor unit serves an internal webpage accessible via port 80. To access the content open the unit's IP in your LAN network on a computer or a phone. The webpage served is as follows.

The JSON link points to a JSON formatted data source, that can be polled periodically to access the uRADMonitor unit readings. As this is done directly by connecting to the uRADMonitor unit, the server compensation layer is not used, so you would receive the raw readings. This is not the preferred way, and additional compensation must be implemented (eg. Temperature offset to compensate for internal heating, other corrections, etc). This functionality is offered rather for debugging and decentralized operation in critical situations such as server failure or malfunction.

- **Data access via the Server RESTful API**

This is the preferred data access method. REST API does not require the client to know anything about the structure of the API. Rather, the server needs to provide whatever information the client needs to interact with the service. An HTML form is an example of this: The server specifies the location of the resource, and the required fields. The browser doesn't know in advance where to submit the information, and it doesn't know in advance what information to submit. Both forms of information are entirely supplied by the server. Lookups should use GET requests. PUT, POST, and DELETE requests should be used for creation, mutation, and deletion.

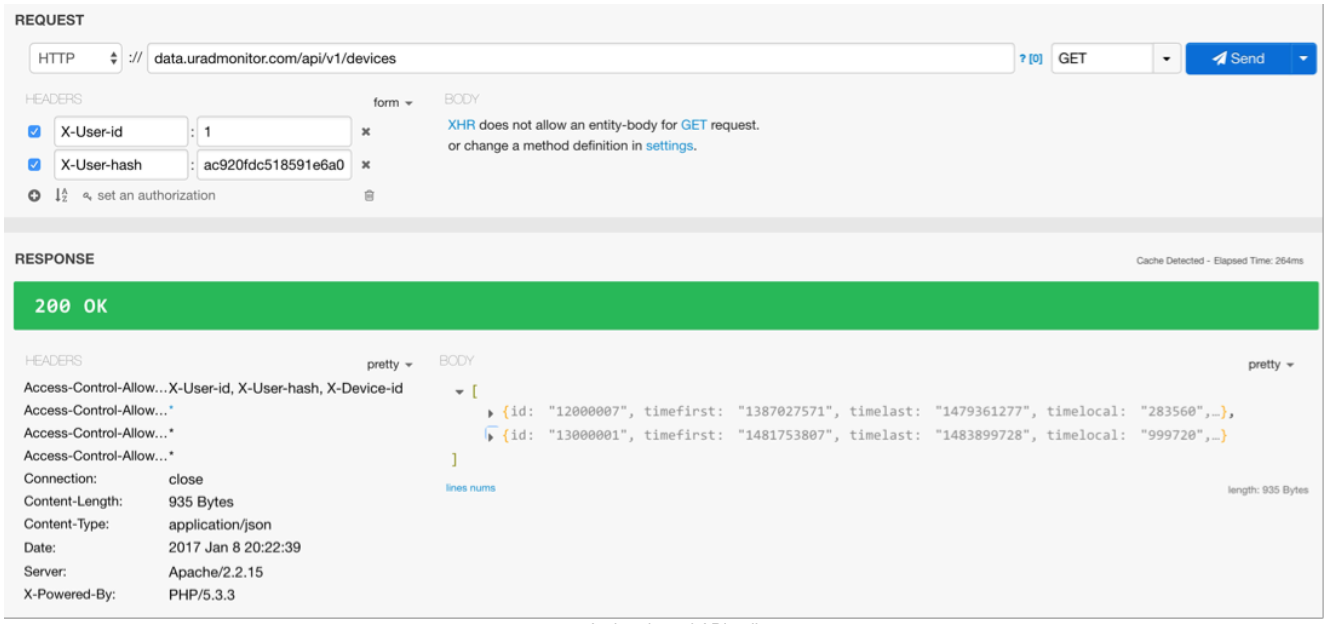
The API is called for both directions of data transfer (upload and download). The uRADMonitor devices use the API to upload their measurements to the server, for further processing and storage in the database. The API is then used to access data by the frontend, the mobile app or third party systems that need the uRADMonitor data.

Server API: Authentication

Some API calls require authentication with user ID and user Key and will return results depending on the privileges and settings of the given user. To authenticate a call, the HTTP GET header must contain two custom fields, defined as follows:

X-User-id	Will contain the user ID.
X-User-hash	Will contain the user Key.

Both the user ID and the user Key are displayed in the Dashboard. Here is call example, using the authentication headers:



Authenticated API call

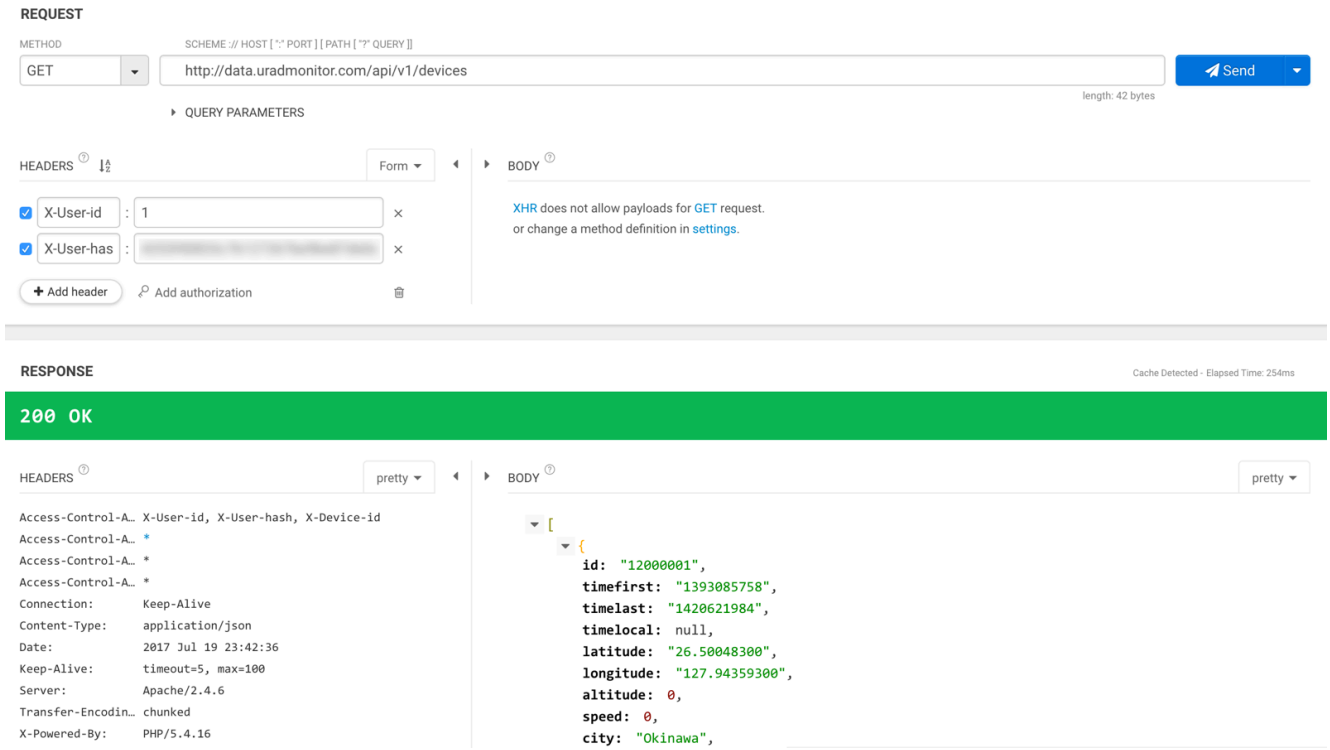
Below the list of API calls is presented. Those that require authentication will be marked accordingly.

Server API: API Calls for data access

For the uRADMonitor RESTful API, there is a common base url, defined as http://data.uradmonitor.com/api/v1/ followed by the following verbs:

1	*devices*	**Full URL**: https://data.uradmonitor.com/api/v1/devices
Method: HTTP GET		**Purpose**: data access
Description		Used to retrieve the list of uRADMonitor units assigned to the user account. The list includes the units the user is either set as owner or has global access to them.
Authentication		yes, using X-User-id and X-User-hash in HTTP Get header

Call example:



Return: summary array of uRADMonitor units in JSON format.

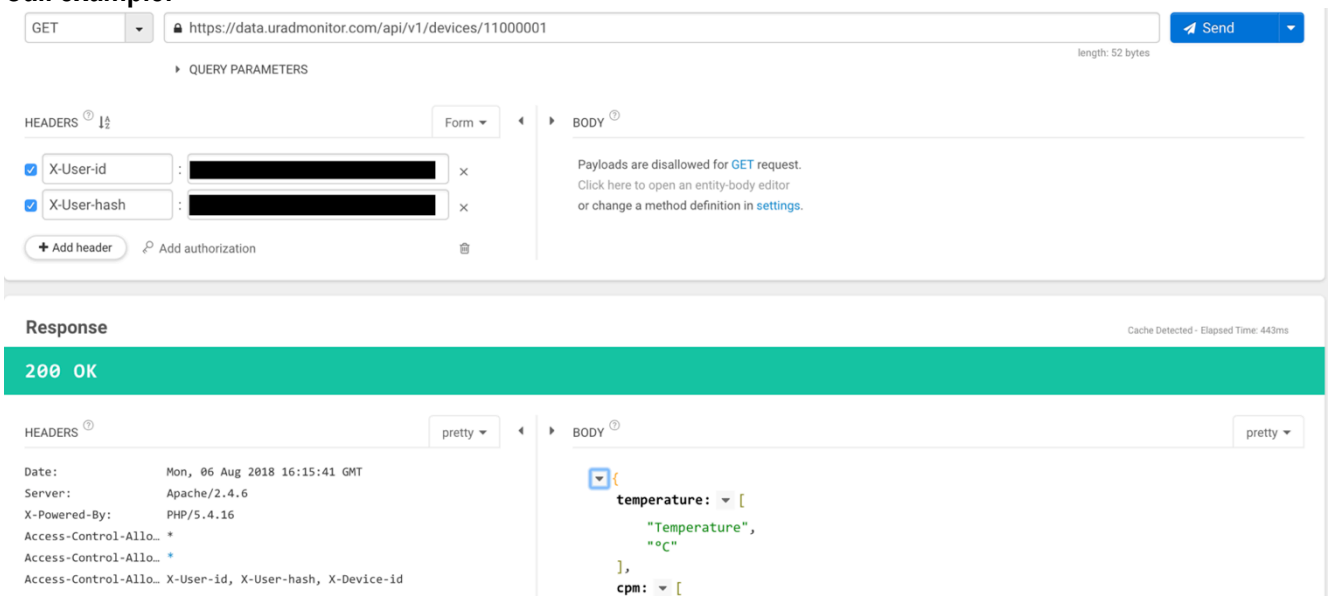
[{ **id :** "82000050", **timefirst :** "1476801965", **timelast :** "1499877474", **timelocal :** "120", **latitude :** "37.46906600", **longitude :** "-79.21035800", **altitude :** 213, **speed :** 0, **city :** "Lynchburg", **country :** "US", **versionsw :** "122", **versionhw :** "103", **status :** null, **mobile :** null, **detector :** "SI29BG", **factor :** 0.01, **avg_temperature :** "25.39", **avg_pressure :** "99268", **avg_humidity :** "67.13", **avg_voc :** "2669238", **min_voc :** "73049", **max_voc :** "11818108", **avg_co2 :** "514", **avg_ch2o :** "0.00", **avg_pm25 :** "950", **avg_noise :** "0.00", **avg_cpm :** "11.40", **avg_voltage :** "380.97", **avg_duty :** "219.68"}, {...}]

Each result in the array contains the following information:

id	the unique uRADMonitor unit ID
timefirst	unix timestamp containing the moment in time the unit first transmitted data
timelast	unix timestamp containing the moment in time of the last data transmission
timelocal	timestamp containing the number of seconds elapsed since the unit was last rebooted
latitude	latitude coordinate in decimal format
longitude	longitude coordinate in decimal format
altitude	altitude coordinate in meters
speed	unit speed in km/h
city	define base city for this unit
country	2 letter country code for the location of this unit
versionsw	firmware version
versionhw	hardware iteration version
status	1 if the unit is online, NULL if it is offline
mobile	1 if the unit is a mobile unit (eg. Model-D units or A3 units installed in buses)
detector	name of radiation detector sensor if the unit has such capabilities (only for Model A, KIT1, D and A3 up to v105)
factor	CPM to Eq Dose Rate linear approximation conversion factor (dependent on "detector")
avg_XX	last 24hours average of the given sensor. Each unit model has a different number of avg_XX values returned, depending on its capabilities and the number of parameters it measures

2	*devices/[ID]*	**Full URL**: https://data.uradmonitor.com/api/v1/devices/[ID]
Method: HTTP GET		**Purpose**: data access
Description		ID is a unique uRADMonitor unit ID (eg. 110000AB) . This call is used to return the list of sensors of the specified unit.
Authentication		yes, using X-User-id and X-User-hash in HTTP Get header

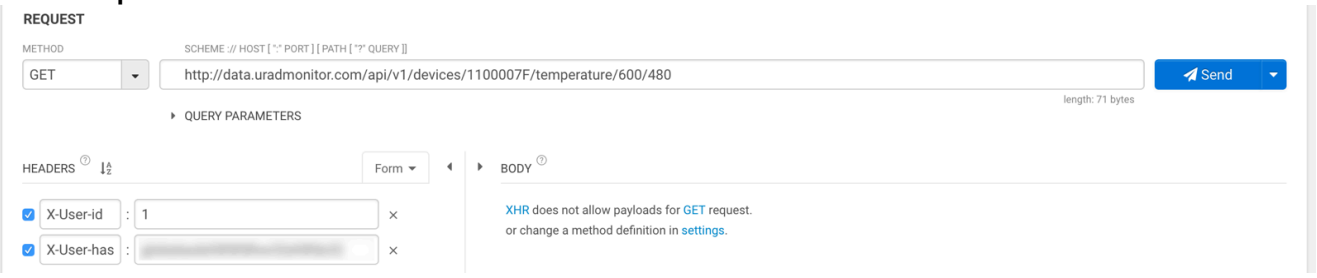
Call example:



Return: list of supported sensors as an array in JSON format, including the unit of measure:
{ temperature : ["Temperature","°C"], cpm : ["Radiation","cpm"], voltage :["Voltage","V"], duty : ["Duty cycle","‰"], all : ["All",""]}

3	*devices/[ID]/[sensor]/ [startinterval]/ [stopinterval]*	**Full URL**: https://data.uradmonitor.com/api/v1/devices/[ID]/[sensor]/[startinterval]/[stopinterval]
Method: HTTP GET		**Purpose**: data access
Description		ID is a unique uRADMonitor unit ID (eg. 110000AB) . Sensor is a sensor name (eg. temperature) or you can also use the special keywork "all" to access data from all sensors installed on the unit. Startinterval is the the number of seconds from the moment of the present to get data from; "stopinterval" is optional and it represents the number of seconds from the moment of present to get data to. If "stopinterval" is not specified, the moment of present is used as the stop point. If there is no data for the query specified, you will receive an empty JSON array.
Authentication		yes, using X-User-id and X-User-hash in HTTP Get header

Call example:

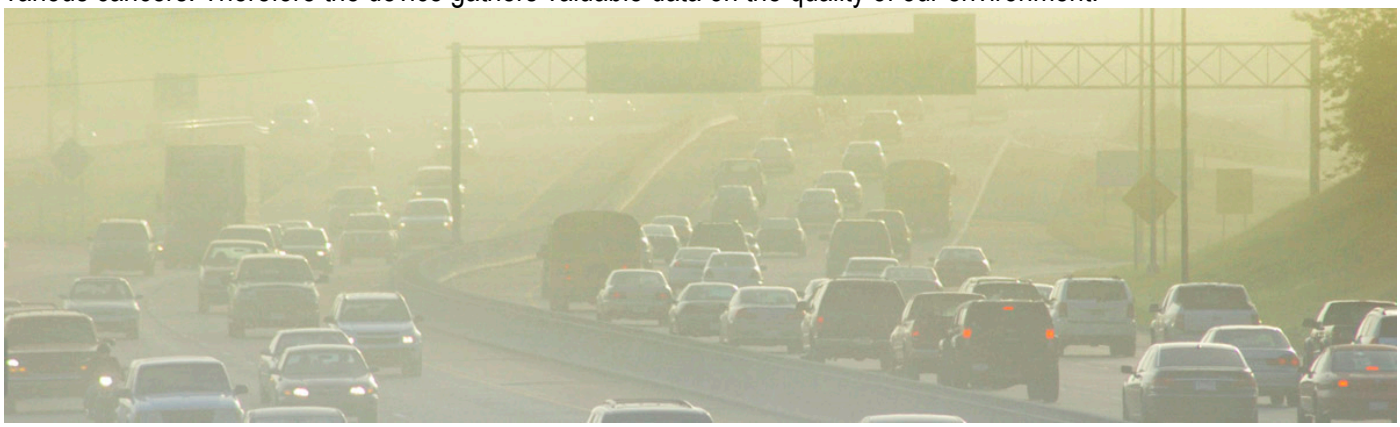


Return: For the previous example call, we receive two temperature measurements, because we specified an interval of 120 seconds and the unit resolution was 1 minute:
[{ **time :** "1500498412", **latitude :** "61.11200000", **longitude :** "-149.90440000", **altitude :** "250.00", **temperature :** "22.00"},
{ **time :** "1500498472", **latitude :** "61.11200000", **longitude :** "-149.90440000", **altitude :** "250.00", **temperature :** "21.93"}]

Additional information is presented under the API tab in the uRADMonitor dashboard:
https://www.uradmonitor.com/dashboard/

Health impact

Many of the parameters measured by Model AIR can have a negative health impact, ranging from simple allergies to various cancers. Therefore the device gathers valuable data on the quality of our environment.



Particulate matter PM2.5 refers to small particles with a diameter of up to 2.5 microns. These particles can penetrate deep into the lungs , causing allergies, respiratory and cardiovascular diseases [1]

[1] Health and Environmental Effects of Particulate Matter (PM), US Environmental Protection Agency